Stabilization and Acceleration of Algebraic Multigrid Method
Recursive Projection Algorithm

A. Jemcov   J.P. Maruszewski

Fluent Inc.

October 24, 2006
1. Motivation
   - Need for Algorithm Stabilization and Acceleration

2. Iterative Methods for Linear Systems
   - Basic Iterative Methods
   - Algebraic Multigrid Method
   - Recursive Projection Method

3. Conclusions
   - Summary and Future Work
Motivation
- Need for Algorithm Stabilization and Acceleration

Iterative Methods for Linear Systems
- Basic Iterative Methods
- Algebraic Multigrid Method
- Recursive Projection Method

Conclusions
- Summary and Future Work
Motivation

Need for Algorithm Stabilization and Acceleration

Iterative Methods for Linear Systems

Basic Iterative Methods
Algebraic Multigrid Method
Recursive Projection Method

Conclusions

Summary and Future Work
1 Motivation
   - Need for Algorithm Stabilization and Acceleration

2 Iterative Methods for Linear Systems
   - Basic Iterative Methods
   - Algebraic Multigrid Method
   - Recursive Projection Method

3 Conclusions
   - Summary and Future Work
Computational science and CFD in particular is characterized by increasingly larger and more physically complex problems.

Today’s computational problems range from a few thousand to a hundreds of millions of unknowns.

Massively parallel computer clusters are routinely used to run those large problems.

Many examples from of large problems in CFD, structural analysis, multi-body dynamics, optimization, chemistry, statistical physics ···

Many times implicit solvers are used to run problems, resulting in large system of linear equations.
Given the need to solve large linear systems of equations, very efficient iterative linear solvers are required. Direct solvers are out of question due to large memory requirements. Two families of iterative solvers are mainly used: stationary and Krylov space linear solvers. Stationary solvers include Gauss-Seidel, Jacobi, ILU(n), etc. Krylov space methods include CG, BCGSTAB, GMRES, TFQMR, etc. Very rarely these solvers are used without some preconditioning and/or acceleration techniques.
Traditionally, in CFD field stationary iterative methods are preferred due to their simplicity and low memory requirements.

Stationary methods are also commonly referred to as relaxation methods or smoothers.

It is well known fact that smoothers have initially fast and then later convergence rates.

As a remedy to this problem, Multigrid strategies of acceleration are commonly used.

Excellent convergence rates were obtained with the use of Multigrid and basic smoothers in many problems.

However, due to their reliance on simple iterative methods, Multigrid methods tend have slower convergence or are even known to diverge when underlying smoother cannot handle system matrix.
Two important undesirable behaviors of linear solvers can be identified:
- Linear solver is converging but the convergence rate is slow
- Linear solver is diverging

Obviously, the second type of behavior is clearly more dangerous because it will not produce meaningful solution.

However, even the first type of the behavior is undesirable in the case of large systems of linear equations because of required time to produce the solution.

Important questions is: "Can something be done to avoid divergence and/or accelerate convergence?"

In this presentation we will see that Recursive Projection Method can enhance AMG Method to produce desirable behavior.
1 Motivation
   - Need for Algorithm Stabilization and Acceleration

2 Iterative Methods for Linear Systems
   - Basic Iterative Methods
     - Algebraic Multigrid Method
     - Recursive Projection Method

3 Conclusions
   - Summary and Future Work
We are interested in solving the following linear system

$$Ax = b$$

- $A$ is a quadratic $N \times N$ real matrix obtained by discretizing some PDE so that it has M-matrix properties, i.e. it is diagonally dominant and diagonal entries are positive whereas off-diagonal entries are non-positive.
- $x$ is a vector of dimension $N$ that contains unknown variables.
- $b$ is a right-hand-side vector of dimension $N$ that contains forcing terms.
Basic iterative methods assume that there is a regular splitting of matrix \( A \) such that

\[ A = M - N \]

Matrix \( M \) is sometimes called preconditioning matrix whereas matrix \( N \) is called defect matrix.

In the case of M-matrices, this splitting leads to convergent splittings.

With this notation, we have

\[ Mx = Nx + b \]

How do we get an iterative method out of this splitting?
Introduce the next iterate $x^{(i+1)}$ and the current iterate $x^{(i)}$ into the splitting

$$Mx^{(i+1)} = Nx^{(i)} + b$$

Solving for $x^{(i+1)}$ we obtain the following iterative algorithm

$$x^{(i+1)} = (M^{-1}N)x^{(i)} + M^{-1}b$$

$$x^{(i+1)} = Rx^{(i)} + M^{-1}b$$

Matrix $R$ is called iteration matrix and this particular form of iterative process is called fixed-point iteration

$$x^{(i+1)} = F(x^{(i)})$$
Stationary Iterative Methods

What properties matrices \( M \) and \( N \) should have?
- They have to produce an iterative method that will converge for arbitrary initial guess \( x^{(0)} \)
- Matrix \( M \) should be easy to invert

Depending on the choice of the splitting, different iterative methods result

Introduce the following splitting of the matrix \( A \)

\[
A = D - L - U
\]

\( D \) is diagonal, \(-L\) is strictly lower and \(-U\) is strictly upper triangular matrix
Stationary Iterative Methods

- Jacobi iterative method
  \[ R = D^{-1} (L + U) \]

- Weighted Jacobi iterative method
  \[ R = (1 - \omega)I + \omega D^{-1} (L + U) \]

- Gauss-Seidel iterative method
  \[ R = (D - L)^{-1} U \]
Examine the propagation of error in iterative process

\[ e^{(i)} = x^{(i)} - x^* \]

Also take into account that for exact solution \( x^* \)

\[ x^* = R x^* + M^{-1} b \]

Error propagation equation

\[ e^{(i+1)} = R e^{(i)} \]

The following recursive property can be established

\[ e^{(m)} = R^m e^{(0)} \]
Formally

\[ \| R^{(m)} \| = \| \lambda(R) \| \leq \| R \|^{m} \| e^{(0)} \| \]

Error is forced to zero if \( \| R \| < 1 \)

\[ \lim_{m \to \infty} R^{m} = 0 \text{ iff } \rho(R) < 1 \]

Here \( \rho(R) \) is a spectral radius

\[ \rho(R) = \max \| \lambda(R) \| \]

Asymptotic convergence rate

\[ m \geq -\frac{d}{\log_{10}[\rho(R)]} \]
Performance of Stationary Iterative Methods

- Asymptotic convergence rate states that for eigenvalues close to zero error is reduced quickly.
- For eigenvalues close to 1, error reduction is very slow.
- For eigenvalues greater than 1 error is amplified.
- Consider simple one-dimensional elliptic model problem

\[ \frac{d^2 x}{d\xi^2} + \sigma x(\xi) = b(\xi) \]

- Discretize problem with finite differences

\[ -x_{j-1} + \frac{2x_j - x_{j+1}}{h^2} + \sigma x_j = b(\xi_j) \]
Performance of Stationary Iterative Methods

- Resulting matrix $\mathbf{A}$ has tridiagonal structure with
  \[
  \mathbf{D} = \text{diag}(2 + \sigma h^2) \\
  \mathbf{L} = \text{tril}(-1) \\
  \mathbf{U} = \text{triu}(-1)
  \]
  - Eigenvalues of weighted Jacobi iteration matrix
    \[
    \lambda_k = 1 - 2\omega \sin^2\left(\frac{k\pi}{n}\right), \quad 1 \leq k \leq n - 1
    \]
  - Eigenvalues of Gauss-Seidel iteration matrix
    \[
    \lambda_k = \cos^2\left(\frac{k\pi}{n}\right), \quad 1 \leq k \leq n - 1
    \]
Performance of Stationary Iterative Methods

- From the asymptotic convergence rate, it is clear that eigenvalues close to 1 will be the ones determining the speed of convergence.

- In principle, high frequency and low frequency content can be identified and spectrum can be divided based on the wave number $k$.

- High frequency spectral content associated with eigenvalues close to zero are efficiently removed by basic iterative methods.

- Low frequency spectral content (smooth error) is difficult to remove with basic iterative methods.

- As the mesh is refined, more and more eigenvalues close to one are introduced slowing down convergence.
Motivation
Fixed Point Methods
Conclusions

Basic Methods
AMG
RPM

Performance of Stationary Iterative Methods

- Jacobi iterations for 2D conduction matrix $5 \times 5$

![Residuals](image1)

![Iteration Matrix Spectrum](image2)

A. Jemcov, J.P. Maruszewski
Recursive Projection Algorithm
Jacobi iterations for 2D conduction matrix $10 \times 10$
Jacobi iterations for 2D conduction matrix $15 \times 15$
Performance of Stationary Iterative Methods

- Jacobi iterations for 2D conduction matrix $20 \times 20$

![Residuals](image1)

![Iteration Matrix Spectrum](image2)

A. Jemcov, J.P. Maruszewski

Recursive Projection Algorithm
Performance of Stationary Iterative Methods

- Jacobi iterations for 2D conduction matrix $25 \times 25$

![Residuals](image1)

![Iteration Matrix Spectrum](image2)
1 Motivation
   - Need for Algorithm Stabilization and Acceleration

2 Iterative Methods for Linear Systems
   - Basic Iterative Methods
   - Algebraic Multigrid Method
   - Recursive Projection Method

3 Conclusions
   - Summary and Future Work
Given the poor performance of basic iterative methods on large grids, the question is if anything can be done to speed them up?

Recall that the smooth part of the error that is associated with eigenvalues close to 1 is responsible for slow rates of convergence.

Can the linear system $Ax = b$ be transformed somehow so that iteration matrix has more favorable spectrum?

Algebraic Multigrid Method provides a strategy that utilizes Galerkin projection in order create equivalent linear system thus transforming the smooth error into the high frequency error.
Two key concepts in AMG are the concept of Algebraic Smoothness and the concept of the Strong Influence

Loosely speaking, algebraically smooth error satisfies the following equation

$$Ae \approx 0$$

This means that smooth error has a relatively small residuals

If $e$ is a smooth error, then components $e_i$ can be approximated well by a weighted average of neighbors through process called agglomeration

$$a_{ii}e_i = - \sum_{j \neq i} a_{ij}e_j$$
Acceleration of Basic Iterative Methods

- Strong influence concept assumes that there is a threshold value \( 0 < \theta \leq 1 \) which can be used to establish the influence of other variables in a given matrix row:

\[
-a_{ij} \geq \theta \max_{k \neq i} (-a_{ik})
\]

- This criterion allows separation of equations in different sets: coarse and fine.

- Interpolation coefficients can be computed \( \omega_{ij} \) so that the error is given by:

\[
e_i = \sum_j \omega_{ij} e_j
\]

- Therefore, value of the error at a particular location can be interpolated from the errors in the neighborhood with a weight coefficients that were chosen to follow smooth error...
Acceleration of Basic Iterative Methods

- Once the splitting on coarse and fine equations based on strong influence principle is known, and once interpolation coefficients $\omega_{ij}$ are computed, it is possible to construct the restriction matrix $\mathcal{R}$.
- Similarly, prolongation matrix $\varphi = \mathcal{R}^T$ can be obtained.
- Restriction of fine matrix to the coarse one:
  \[ A^{2h} = \mathcal{R} A^h \varphi \]
- Solve for corrections on each coarse level:
  \[ A^{nh} e^{nh} = -r^{nh} \]
- Correct fine solution:
  \[ x^h \leftarrow x^h + e^{2h} \]
V-cycle AMG strategy
At each multigrid level matrix $A$ is restricted through Galerkin projection, coarse corrections were computed and fine solution corrected through prolongation operator.
Correction were solved by basic iterative methods (smoothers)
AMG Performance

- AMG with Gauss-Seidel smoother, 5 × 5 grid

![Graph showing AMG performance with Gauss-Seidel smoother](graph.png)
AMG Performance

- AMG with Gauss-Seidel smoother, 10 × 10 grid
AMG Performance

- AMG with Gauss-Seidel smoother, $15 \times 15$ grid

![Graph showing AMG performance](image)

A. Jemcov, J.P. Maruszewski

Recursive Projection Algorithm
AMG Performance

- AMG with Gauss-Seidel smoother, 20 × 20 grid
AMG Performance

- AMG with Jacobi smoother, $20 \times 20$ grid

![Graph showing AMG Performance](image)

- AMG with Jacobi smoother, 20 $\times$ 20 grid
Outline

1. Motivation
   - Need for Algorithm Stabilization and Acceleration

2. Iterative Methods for Linear Systems
   - Basic Iterative Methods
   - Algebraic Multigrid Method
     - Recursive Projection Method

3. Conclusions
   - Summary and Future Work
Stabilization and Acceleration of AMG

- How important is the fact that the underlying matrix had M-matrix properties?
- Consider again the same 2D conduction matrix but this time change off-diagonal coefficients to violate M-matrix property

\[ a_{5,10} = a_{10,5} = -6.5 \]

- Before the change, those coefficients had a value of \(-1\)
- There is no change to the structure of the matrix on to the value of coefficients
Stabilization and Acceleration of AMG

- AMG with Gauss-Seidel with $5 \times 5$ non-M-matrix
Clearly, one eigenvalue has left the unit circle and no convergent splitting \( A = M - N \) based on simple iterative methods is possible.

Spectral radius of iteration matrix \( R \) is greater than 1, iterations will diverge.

Since AMG is just a strategy that uses basic iterative methods and sequence of coarse linear systems to accelerate convergence, it cannot produce a convergent solution.

What can be done to keep AMG from diverging?
Stabilization and Acceleration of AMG

- One way of keeping the things from falling apart is to use another iterative method that doesn’t suffer from this limitation.
- Krylov space solvers can be used but this is a departure from AMG concept.
- Another possibility is to remove unwanted eigenvalues from the spectrum and then proceed with AMG.
- This is the approach taken by Recursive Projection Algorithm (RPM).
In principle, eigenvalues outside the unit disk are associated with unstable part of the solution $\mathbf{x}$

It is possible to associate a subspace called unstable space $\mathcal{P}$ with these eigenvalues so that we could project the problem onto stable and unstable parts.

In that case, solution vector will be split into stable $\mathbf{q}$ and unstable part $\mathbf{p}$

$$\mathbf{x}^{(i)} = \mathbf{p}^{(i)} + \mathbf{q}^{(i)}$$

$\mathbf{p} \in \mathcal{P}$, $\mathbf{q} \in \mathcal{Q}$

This process of separation of the solution vector $\mathbf{x}$ is achieved by a process of projection
Stabilization and Acceleration of AMG

- Define subspace $\mathcal{P}$ and $\mathcal{Q}$ of $\mathbb{R}^N$ by
  - $\mathcal{P}$ is the maximal invariant subspace of $\mathbb{R}$
  - $\mathcal{Q} = \mathbb{R}^N - \mathcal{P}$, the orthogonal complement of $\mathcal{P}$
- Introduce projectors $\mathbf{P}$ and $\mathbf{Q}$ so that:
  \[
  \mathbb{R}^N = \mathbf{P}\mathbb{R}^N \oplus \mathbf{Q}\mathbb{R}^N
  \]
  \[
  \mathbf{x}^{(i)} = \mathbf{p}^{(i)} + \mathbf{q}^{(i)}
  \]
  \[
  \mathbf{p} = \mathbf{P}\mathbf{x} \in \mathcal{P} \quad \mathbf{q} = \mathbf{Q}\mathbf{x} \in \mathcal{Q}
  \]
- Project the fixed-point problem onto $\mathcal{Q}$ and $\mathcal{P}$:
  \[
  \mathbf{x}^{(i+1)} = \mathbf{F}(\mathbf{x}^{(i)}) = \mathbf{R}\mathbf{x}^{(i)} + \mathbf{M}^{-1}\mathbf{b}
  \]
  \[
  \mathbf{p} = \mathbf{P}\mathbf{F}(\mathbf{x}^{(i)}) = \mathbf{f}(\mathbf{x}^{(i)}) \quad \mathbf{q} = \mathbf{Q}\mathbf{F}(\mathbf{x}^{(i)}) = \mathbf{g}(\mathbf{x}^{(i)})
  \]
Stabilization and Acceleration of AMG

- These projectors provide natural problem decomposition of the iterative procedure:

  \[ p^{(i+1)} = f(p^{(i)}) \quad q^{(i+1)} = g(q^{(i)}) \]

- The new iterative procedure can be devised:

  \[(I - f_p) (p^{(i+1)} - p^{(i)}) = f(p^{(i)}, q^{(i)}) - p^{(i)}\]

  \[q^{(i+1)} = g(p^{(i)}, q^{(i)})\]

  \[x^{(i)} = p^{(i)} + q^{(i)}\]

- Here, Newton’s method is used on \(f(p^{(i)})\) and the AMG iterations on \(g(q^{(i)})\)
Stabilization algorithm:

1. \( p^{(0)} = Px^{(0)} \quad q^{(0)} = Qx^{(0)} \)
2. Do until convergence:
   - \( p^{(i+1)} = p^{(i)} + (I - f_p)^{-1} \left( f(p^{(i)}, q^{(i)}) - p^{(i)} \right) \)
   - \( q^{(i+1)} = g(p^{(i)}, q^{(i)}) \)
3. \( x^{(M)} = p^{(M)} + q^{(M)} \)
In order to isolate eigenvalues outside of the unit circle, projectors $P$ and $Q$ must be determined. If $Z$ is an orthonormal basis of for subspace $\mathcal{P}$, then projectors are defined as:

$$P = ZZ^T$$

$$Q = I - ZZ^T$$

The question is how to compute this basis inexpensively?

Could we use current iterates to estimate the basis?
Recall the recurrence relation

\[ e^{(i+1)} = Re^{(i)} \]

Apply recursion to the stable part

\[ \Delta q^{(i+1)} = R \Delta q^{(i)} \]

Here differences \( \Delta q^{(i)} \) are used to estimate dominant eigenspace through Gram-Schmidt procedure.

In practice we look into two last differences \( \Delta q^{(i+1)} \) and \( \Delta q^{(i)} \) and then perform orthogonalization procedure to estimate basis \( Z \).

Once the basis \( Z \) is known, projectors can be determined.
Given the projectors $P$ and $Q$, unstable space can be extracted and the modified iterative procedure can be applied.

Eigenvalues are estimated from the diagonal elements of the upper triangular matrix in $QR$-decomposition.

Currently only two differences of iterates $\triangle q^{(i)}$ are used at the time to approximate unstable space.

The basis $Z$ is recursively enlarged by adding one or two orthogonal vectors at the time until we obtain sufficient approximation of the unstable space.
RPM Performance

- AMG with Gauss-Seidel with $5 \times 5$ non-M-matrix

![Graph showing residual vs iteration for AMG and Iteration Matrix Spectrum](image)
RPM Performance

- RPM with Gauss-Seidel with $5 \times 5$ non-M-matrix

Recursive Projection Method

Stabilized Iteration Matrix Spectrum

A. Jemcov, J.P. Maruszewski
Recursive Projection Algorithm
AMG with $400 \times 400$ anisotropic pressure matrix
RPM Performance

- RPM with $400 \times 400$ anisotropic pressure matrix

**Recursive Projection Method**

**Stabilized Iteration Matrix Spectrum**

A. Jemcov, J.P. Maruszewski

Recursive Projection Algorithm
RPM Performance

- AMG with $59 \times 59$ FEM shell matrix

A. Jemcov, J.P. Maruszewski
Recursive Projection Algorithm
RPM Performance

- RPM with $59 \times 59$ FEM shell matrix

Graphs showing:
- Recursive Projection Method
- Stabilized Iteration Matrix Spectrum
FVM Pressure Matrix
Matrix Dimension $N = 1729830$, Non-zero entries $n = 9131220$

- AMG
- RPM-AMG

Recursive Projection Algorithm
Motivation

Need for Algorithm Stabilization and Acceleration

Iterative Methods for Linear Systems

Basic Iterative Methods
Algebraic Multigrid Method
Recursive Projection Method

Conclusions

Summary and Future Work
RPM has excellent convergence properties
It enables other methods such as AMG to perform outside of their range of applicability
It is implemented as a wrapper to existing solver so it reuses the code
Future work will concentrate on application of RPM to non-linear solvers (fixed point methods)